

Real-time Astrometry using Phase Congruency

Andrew Lambert

School of Engineering and IT, UNSW Canberra, Canberra, Australia

YongYi Tang

South China University of Technology, China

Manuel Cegarra Polo

School of Engineering and IT, UNSW Canberra, Canberra, Australia.

ABSTRACT

Phase congruency is a computer vision technique that proves to perform well for determining the tracks of optical objects. We report on a real-time implementation of this using an FPGA and CMOS Image Sensor, with on-sky data. The lightweight instrument can provide tracking update signals to the mount of the telescope, as well as determine abnormal objects in the scene.

1. INTRODUCTION

The ability to track a satellite or Resident Space Object (RSO) using an optical telescope is an exercise in prediction, search, detection, and tracking. The success in all of these phases relies on accurate positioning and motion of the telescope mount, as the field of view of the telescope is usually narrow, only allowing for a few exposures of the object, at best, before it disappears from view. During this time the mount must be directed and accelerated to rate-track the object. At Low Earth Orbit (LEO) the total time available for a horizon-to-horizon track is generally less than 10 minutes, so any engineering to support the observation by improving the accuracy of direction and rate estimation gives a significant advantage to the timely completion of the phases.

Further complicate this with the likely small return of light from either the active ranging source, or Sun illumination, then longer exposures are required to detect the RSO, and hence surrounding objects such as the stellar traces are the only cues for correcting the telescope performance. These traces are created as the stars in the field of view appear to travel at different rates to the RSO and hence smear across the imagery. Alternatively the object to be found may also form a trace of different length to the celestial objects, and these may be useful for astrometric observations that feed the prediction for subsequent observations. An example trace is shown in Figure 1.

The prediction phase is usually enabled from previous observations, usually globally separated, that are provided in the Space Track or Celestrak catalogues, for example. However, particularly for LEO, these have a significant accrued error since previous observations, and these must be reacted to in the search phase of the observation, albeit with the initial information.

Hence there is a need for real-time observation and computation for astrometry that will enable further observations by enabling the prediction, and in a shorter term, correction of the optical system pointing or tracking. Computer vision techniques have been proposed for this exercise, including edge and corner detectors to find the start and end of “tracklets”, and for finding the RSO of interest in the image. An interesting algorithm also proposed[1] was Phase Congruency, that on its foundation seems ideal for finding these features in a noisy image, and for implementing in real-time.

The work presented here details the real-time implementation of the Phase Congruency algorithm in a Field Programmable Gate Array (FPGA). It includes refinements to allow this to work at pixel rates of up to 250 MHz (50 MHz) for our current CMOS image sensor and FPGA, developed for adaptive optics solutions[2]. This pixel rate can be made higher with improved hardware. These pixels rates define the frame-rate, which must of course also satisfy the light budget and exposure time associated with the tracklets and RSO of interest.

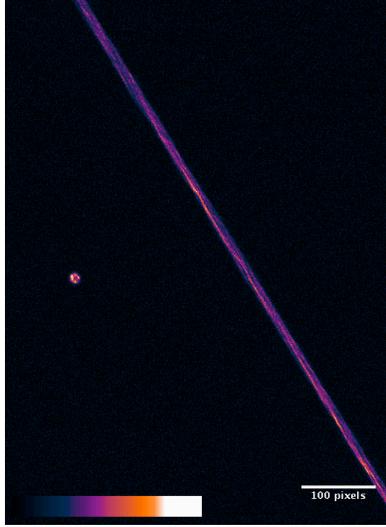


Figure 1: An example RSO (SL-8) and tracklet. This is shown in pseudo color to highlight the effects of turbulence or tracking errors on the integrity of the tracklet line itself. (Image captured with Falcon Telescope Network at UNSW Canberra[3])

2. PHASE CONGRUENCY ALGORITHMS

The concept of Phase Congruency is simple[4], in that at edges, corners, and general regions of fast changing contrast, the Fourier spatial frequency content and its harmonics have a defined phase. For example, a square wave pattern across the image is constructed by $\sin(x')$, $\sin(3x')$, and higher odd harmonics, where the x' coordinate crosses perpendicular to the edge. The associated phase of each sine wave is zero degrees. It has also been noted that at points of symmetry in the image, these harmonics have defined 0 or 180 degree phase relations[5]. Note that the Phase Congruency measure is independent of the contrast level at the edge. As it is unlikely that the square wave pattern extends across the full image, these spatial frequencies are localised to the region around the edge. A phasor representation of this relationship usually shows all phasors adding in a line, and hence the longest phasors in the Fourier spectrum indicate these sharp features. Surrounding noise and other structures, will of course appear as a noise circle to the exactness of this phasor, however its presence is usually higher than the randomised summation where there are less features. Of course, one must explore the direction of the x' coordinate, and its perpendicular counterparts.

For the style of imagery we are proposing the tracklets, RSO, and other features are generally sparsely distributed on or within a noise background, and parallel to each other. This aids the finding of tracklet orientation as most features should be evidenced along corresponding axes in the Fourier spectrum. An adaptive filter running on the Fourier transform of each image would likely find and improve the signal to noise of the estimations by ignoring the regions not on these features. However, attempting this requires a full fast Fourier transform of the image in two dimensions, and may not be reasonably implemented in real-time. Similarly, the filter banks proposed could be adaptively oriented to align to the likely orientations in the image, and gain improvement in signal to noise in the same fashion.

We have not implemented these adaptive components in this work, but they are an obvious progression, and inherently answer the questions of mount orientation and rate for real-time adjustments. Instead we have followed the work published and implemented small two-dimensional log Gabor filters in parallel in the FPGA hardware, with pre-defined operations. In this way streaming data from the image sensor can be run through the odd, H_o , and even H_e filters providing streams of data,

$$\begin{aligned}
 e_n(x, y) &= f(x, y) \otimes H_n^e(x, y) \\
 o_n(x, y) &= f(x, y) \otimes H_n^o(x, y) \\
 A_n^2(x, y) &= e_n^2(x, y) + o_n^2(x, y)
 \end{aligned} \tag{1}$$

from which the measures of the Phase Congruency, symmetry and asymmetry[5] can be obtained,

$$PC_2(x) = \frac{\sum_n W(x) [A_n(x) \Delta\phi_n(x) - T]}{\sum_n A_n(x) + \eta} \quad (2)$$

$$Sym(x) = \frac{\sum_n (|e_n| - |o_n| - T)}{\sum_n A_n(x) + \eta} \quad (3)$$

$$ASym(x) = \frac{\sum_n (|o_n| - |e_n| - T)}{\sum_n A_n(x) + \eta} \quad (4)$$

with minimal (on the order of a few pixels) latency, for subsequent processing. In fact we calculate the revised version of the Phase Congruency equation;

$$PC_2(x) = \frac{\sum_n e_n(x) \bar{\phi}_e(x) + o_n(x) \bar{\phi}_o(x) - |e_n(x) \bar{\phi}_o(x) - o_n(x) \bar{\phi}_e(x)| - T}{\sum_n A_n(x) + \eta} \quad (5)$$

Here, as defined in the original works[4,5], the facto η and the threshold T are used to condition the noise and singularity of the measurement. Presently these are both zero in the implementation, but as we gain more on-sky results we will investigate the adaptive determination of these parameters. Covariance and cross-covariance are also calculated on-the-fly. The instrument now can provide these “live” metrics about each pixel in the image, and subsequent processing can extract the necessary information from the streams. We leave this further extraction and usage for further work.

3. SIMPLIFICATIONS FOR SPEED AND SIZE OF IMAGE

The log Gabor filters detailed above are implemented by delaying the incoming pixel raster through a series of line First-In-First-Out (FIFO) memories, and single pixel delays using latches, to provide each of the necessary multiplicands in each of the parallel two dimensional filters at the same clock edge. This stream based approach provides then the resultant streams for each pixel that arrives, with the latency defined by the dimensions of the filter in lines and pixels (columns). These feeds of delayed pixels could then be directed to the many high precision multipliers in the modern FPGAs, but we take another optimization at this stage to avoid the use of the multipliers. Figure 2 illustrates the parallelism of the architecture implemented here, with the memories providing all the current and delayed pixel data to the filter blocks.

The filters are approximations from different orientations and scales of the log Gabor functions, but could be any similar filter. We use four scale levels and four orientations for these filters, and run 3x3 or 5x5 filters at each stage. The spatial frequency effect of two such filters (two different scales) for four different orientations are shown in Figure 3. The filters are designed to capture different regions of the spatial frequency content, and assemble the phases of harmonics to test the axiom of phase congruency.

It is possible to use the nearest power of two for each filter coefficient over these sizes of filter without biasing the results by too much. When this step is adopted, the multiplication by these filter coefficients, instead becomes a bit shift (and possibly a sign change) to the data being accumulated. We show in Figure 4 the effect of this “quantisation” in the frequency domain for the case of the even and odd filters at the first scale, where it can be seen the actual coefficient and the approximation still adopt phase changes and amplitude masking over similar areas of the spatial frequency domain. Recall these filters are to examine the energy and phase confined to regions of the spectrum, so any similar filter characteristic will provide the same analysis, albeit without orthonormality which is not required since there is no requirement to reconstruct the image. The subjective similarity of both phase and frequency is argued to be sufficient here to enable an extremely efficient and fast method of filter.

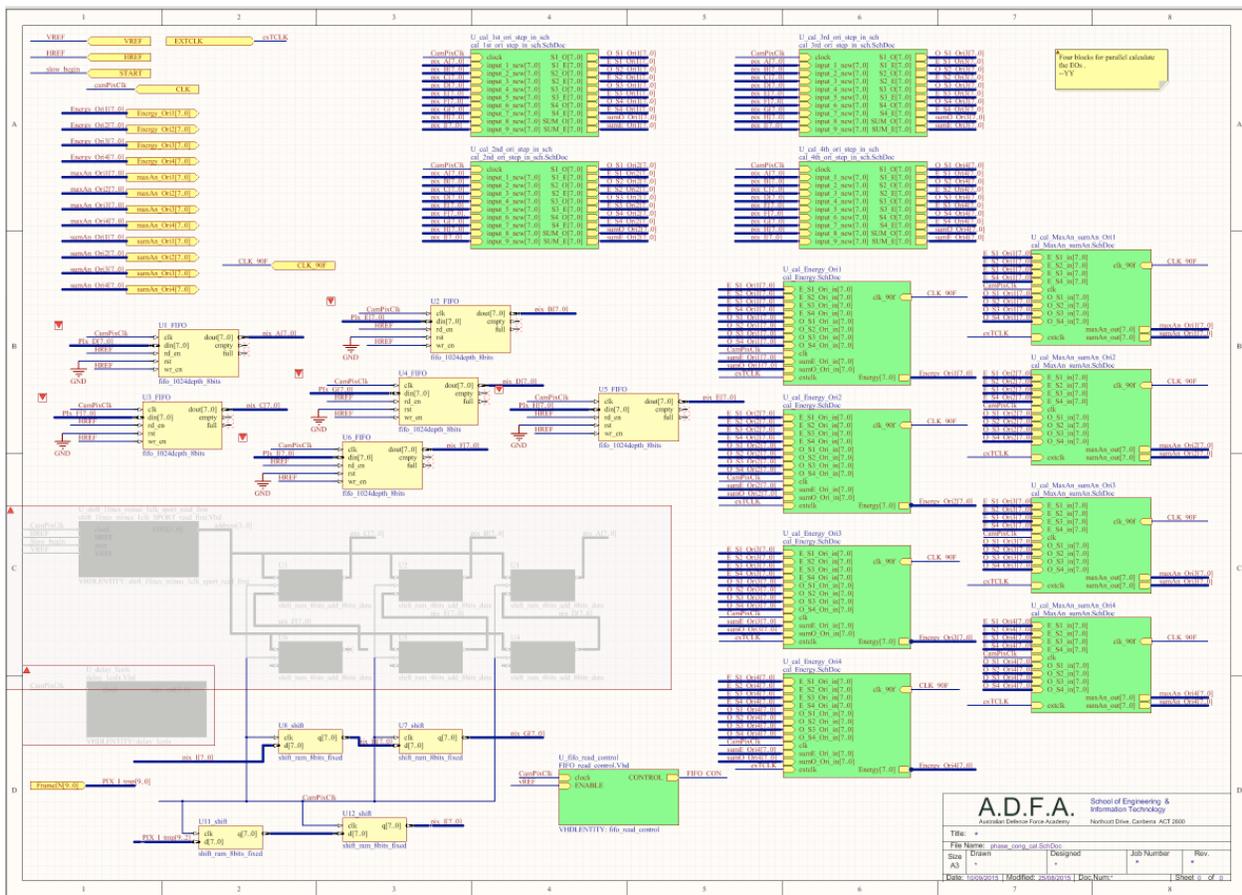


Figure 2: The parallel streaming nature of this implementation are apparent in this sub-sheet of the FPGA design, where pixel streams for a 3x3 filter set in this case are delayed by row and column FIFOs (yellow) and passed in parallel to filter banks (green). Thick lines indicate buses of data, and the multitude of these exemplifies the power of FPGA implementation.

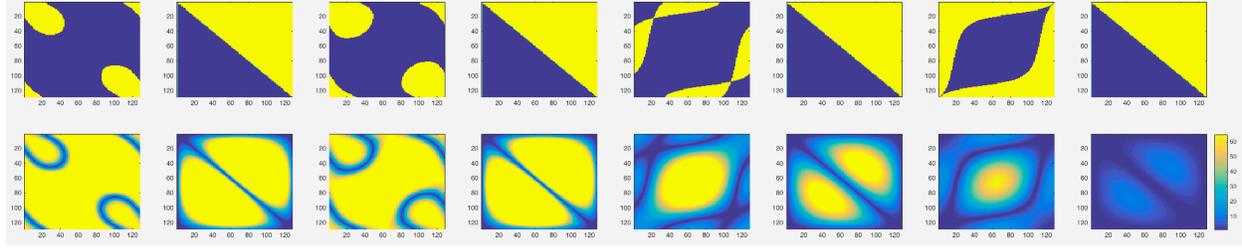


Figure 3: The spatial frequency domain phase (top level) and magnitude (bottom level) for each of the four orientations at two different scales in the filter banks.

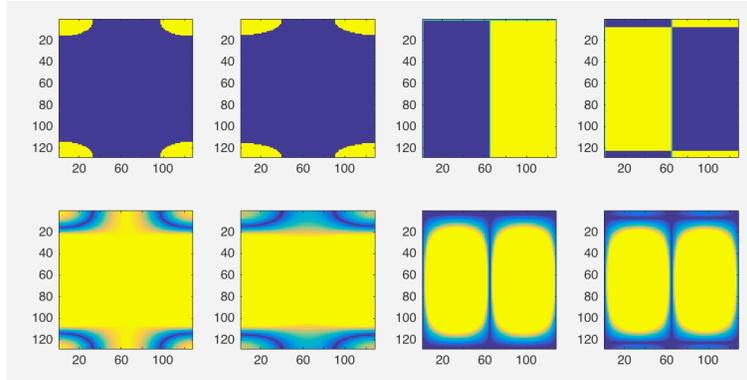


Figure 4: The slight differences in the spatial frequency domain coverage of the quantized filters employed here, is shown for comparison. The Fourier phase (top row) and magnitude (bottom row) are shown for the even (left set) and odd (right set) filters, for the original filter (left most) and the quantized filter (rightmost) in each pair.

4. CONCLUSION

We have reported on the real-time implementation of the Phase Congruency algorithm within an FPGA. The purpose of this is to ascertain in poor signal to noise imagery, the presence, orientation and location of tracklets and RSO for astrometry purposes, and so mount pointing and tracking may be improved. This unit takes its pixel data from an adjacent CMOS image sensor, but could equally well employ a commercial camera solution. The FPGA processes filter banks corresponding to log Gabor wavelet filters at different scales and orientations. These run in parallel and provide the result limited principally to the speed of data transfer within the FPGA. This is possible since the filter coefficient approximations are implemented as bit shifting operations rather than multiplications. The end goal of this work is to provide a real-time astrometry solution for refining the tracking process in a tight feedback loop.

5. REFERENCES

1. Brien Flewelling, Brad Sease, Computer Vision Techniques Applied to Space Object Detect, Track, ID, and Characterize, AMOS, 2014.
2. Andrew Lambert and Manuel Cegarra Polo, Real-time algorithms implemented in hardware for centroiding on a Shack-Hartmann Sensor, *OSA Imaging and Applied Optics Congress*, Arlington, VA, USA, June 2015.
3. Francis K. Chun, Roger D. Tippetts, Michael E. Dearborn, Kimberlee C. Gresham, Ryan E. Freckleton, Martin W. Douglas The U.S. Air Force Academy Falcon Telescope Network, AMOS, 2014.
4. Peter Kovesi, Image Features From Phase Congruency, *Videre: A Journal of Computer Vision Research*. MIT Press. Volume 1, Number 3, Summer 1999.
5. Peter Kovesi, Symmetry and Asymmetry from Local Phase, *Tenth Australian Joint Conference on Artificial Intelligence*, 1997.